(F94) Metodi statistici per l'apprendimento

Introduzione

Docente: *Nicolò Cesa-Bianchi* versione 27 febbraio 2017

I sistemi di apprendimento automatico possono essere utilizzati per risolvere diversi tipi di problemi di inferenza. Per esempio: categorizzare o partizionare dei dati, predire una serie temporale, pianificare una sequenza di azioni. In questo corso ci focalizzeremo su sistemi di apprendimento automatico il cui scopo è imparare delle funzioni che associano un'etichetta y ad un dato $\boldsymbol{x}$. Una volta apprese, queste funzioni possono essere utilizzare per categorizzare un documento o un'immagine. Oppure predire quale annuncio pubblicitario è più probabile venga cliccato dal visitatore di un sito. O anche predire il reddito di un individuo sulla base di indicatori del suo stile di vita. O ancora diagnosticare una malattia sulla base della cartella clinica del paziente. Si noti che le etichette y implicitamente definite in questi esempi sono di due tipi diversi: etichette simboliche, come le categorie di un documento o le malattie e etichette numeriche, come il reddito. Nel primo caso parliamo di un problema di categorizzazione (o classificazione) con insieme di etichette $\mathcal{Y}$ (p.es., $\mathcal{Y} = \{\text{sport}, \text{politica}, \text{spettacolo}\}$). Nel secondo caso parliamo invece di un problema di regressione, dove l'insieme di etichette è contenuto nei reali $\mathbb{R}$. In un problema di classificazione, generalmente gli errori non sono graduati: se prediciamo la categoria sbagliata commettiamo un errore, qualunque sia la classe predetta. In regressione, invece, dove le etichette hanno un valore numerico, l'errore è graduabile a seconda della distanza fra il valore della classe predetta e quello della classe corretta.

Per valutare la bontà di una predizione in un problema di classificazione o regressione si utilizza una **funzione di perdita** non negativa ℓ che misura la discrepanza fra etichetta predetta ed etichetta vera. Se per il dato $\boldsymbol{x}$ l'etichetta corretta è y, la predizione $\widehat{y}$ viene valutata con $\ell(y, \widehat{y}) \geq 0$. In un problema di classificazione la funzione di perdita più tipicamente utilizzata è quella zero-uno:

$$\ell(y, \widehat{y}) = \begin{cases} 0 & \text{se } y = \widehat{y}, \\ 1 & \text{altrimenti.} \end{cases}$$

In alcuni casi, come ad esempio il riconoscimento di mail spam dove $\mathcal{Y} = \{\text{spam}, \text{nonspam}\}$, possiamo penalizzare un falso negativo (ovvero una mail non spam erroneamente classificata come spam) rispetto ad un falso positivo (ovvero una mail spam non classificata come tale). Ad esempio,

$$\ell(y, \widehat{y}) = \begin{cases} 2 & \text{se } y = \text{nonspam e } \widehat{y} = \text{spam}, \\ 1 & \text{se } y = \text{spam e } \widehat{y} = \text{nonspam}, \\ 0 & \text{altrimenti.} \end{cases}$$

In un problema di regressione, invece, funzioni di perdita tipicamente utilizzate sono la perdita assoluta $\ell(y, \widehat{y}) = |y - \widehat{y}|$ e la perdita quadratica $\ell(y, \widehat{y}) = (y - \widehat{y})^2$. Si noti che queste funzioni hanno un significato solo per etichette numeriche e i loro valori crescono all'aumentare della distanza fra y e $\widehat{y}$.

In certi casi, può essere comodo predire in un insieme $\mathcal{Z}$ diverso da $\mathcal{Y}$. Per esempio, si consideri il problema di assegnare una probabilità $\widehat{y} \in (0, 1)$ all'evento $y =$ "domani piove" (e quindi implicitamente assegnare probabilità $1 - \widehat{y}$ all'evento complementare $y =$ "domani non piove"). In questo

caso, $\mathcal{Y} = \{$"domani piove", "domani non piove"$\}$ e $\mathcal{Z} = (0, 1)$. Una funzione di perdita spesso utilizzata per questo problema è quella logaritmica. Identificando l'etichetta "domani piove" con $+1$ e l'etichetta "domani non piove" con -1 abbiamo che

$$\ell(y, \widehat{y}) = \begin{cases} \ln \frac{1}{\widehat{y}} & \text{se } y = +1 \text{ (cioè domani piove),} \\ \ln \frac{1}{1-\widehat{y}} & \text{se } y = -1 \text{ (cioè domani non piove).} \end{cases}$$

Si noti che

$$\lim_{\widehat{y} \to 0^+} \ell(+1, \widehat{y}) = \lim_{\widehat{y} \to 1^-} \ell(-1, \widehat{y}) = \infty \ .$$

Questo in pratica "scoraggia" il predittore da generare predizioni $\widehat{y}$ "troppo certe", ovvero troppo vicine a zero o uno, in quanto essere potrebbero dar luogo a valori arbitrariamente alti della funzione di perdita.

Il dato $\boldsymbol{x}$ è tipicamente un record di una base di dati. In molti casi è possibile codificare il record in un vettore di numeri, un formato che si presta bene a essere analizzato in termini geometrici. Per esempio, tutte le volte che il dato è composto da un insieme di quantità omogenee, come i pixel di un'immagine, è naturale rappresentarlo come un vettore in $\mathbb{R}^d$ (dove d sarebbe il numero di pixel nel caso dell'immagine). In altri casi, questa codifica è un po' più laboriosa. Per esempio, un documento può essere rappresentato come un vettore le cui coordinate sono le parole di un dizionario e i cui valori sono la frequenza con la quale la parola corrispondente alla coordinata appare nel documento. In altri casi ancora la codifica vettoriale può essere forzata. Per esempio, in una cartella clinica tipicamente compaiono attributi numerici non omogenei, come CAP ed età. Oppure attributi simbolici, come il sesso, non rappresentabili su un asse cartesiano. In questo corso tratteremo prevalentemente la situazione in cui il dato può essere naturalmente rappresentato come un vettore di numeri $\boldsymbol{x} \in \mathbb{R}^d$. Quando scriviamo $\boldsymbol{x} \in \mathcal{X}$ significa che non insistiamo sul fatto che $\boldsymbol{x}$ sia un vettore di numeri, ma assumiamo più genericamente che sia un record di una base di dati.

Un classificatore per un problema di classificazione è una funzione $f : \mathcal{X} \to \mathcal{Y}$ (oppure $f : \mathcal{X} \to \mathcal{Z}$ se le predizioni appartengono ad un insieme $\mathcal{Z}$ diverso da $\mathcal{Y}$). Se $\mathcal{Y}$ contiene due sole etichette, per esempio $\mathcal{Y} = \{\text{spam}, \text{nonspam}\}$, allora parliamo di un problema di classificazione binaria e assumiamo convenzionalmente $\mathcal{Y} = \{-1, +1\}$ come abbiamo fatto nell'esempio della pioggia. Analogamente, un regressore per un problema di regressione è una funzione $f : \mathcal{X} \to \mathbb{R}$.

Anche se sono state studiate varie modalità di apprendimento, in questo corso ci concentriamo su una in particolare: la modalità di apprendimento per esempi (o apprendimento supervisionato). Un **esempio** è una coppia $(\boldsymbol{x}, y)$ composta da un dato $\boldsymbol{x}$ e dalla sua etichetta y. L'etichetta y è quella che riteniamo corretta per quel dato. Per esempio, il reddito effettivamente percepito da un individuo, oppure la categoria semantica che un lettore associerebbe ad un documento. Un **training set** è un insieme (o più correttamente multinsieme) $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_m, y_m)$ di esempi. Un algoritmo di apprendimento tramite esempi è un algoritmo che riceve in input un training set e fornisce in output un classificatore oppure un regressore.

Per stimare la capacità predittiva di un classificatore o regressore, che è la cosa alla quale siamo in ultima analisi interessati, si utilizza tipicamente un insieme di esempi chiamato test set. Un test set è un insieme $(\boldsymbol{x}'_1, y'_1), \ldots, (\boldsymbol{x}'_n, y'_n)$ di esempi a cui l'algoritmo di apprendimento non ha accesso.

Dato un classificatore o regressore f, stimiamo la capacità predittiva di f attraverso il **test error**

$$\frac{1}{n}\sum_{t=1}^{n}\ell\big(y'_t, f(\boldsymbol{x}'_t)\big)\ .$$

Il nostro scopo è allora quello di formulare una teoria che ci permetta di sviluppare algoritmi di apprendimento in grado di generare predittori con basso test error.

Dato che l'unica informazione che l'algoritmo riceve in ingresso è il training set $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_m, y_m)$, un approccio ovvio allo sviluppo di algoritmi di apprendimento si basa sull'assunzione che il training error

$$\widehat{\mathrm{er}}(f) = \frac{1}{m}\sum_{t=1}^{m}\ell\big(y_t, f(\boldsymbol{x}_t)\big)$$

di un classificatore o regressore f sia correlato al suo test error.

Sia $\mathcal{F}$ un insieme dato di classificatori o regressori. Il metodo di **minimizzazione del rischio empirico** (ERM, empirical risk minimization) indica l'algoritmo di apprendimento che sceglie la funzione in $\mathcal{F}$ che minimizza il training error,

$$\widehat{f} = \operatorname*{argmin}_{f\in\mathcal{F}} \widehat{\mathrm{er}}(f)\ .$$

Putroppo non è detto che questa strategia funzioni sempre. Per esempio, supponiamo che $\mathcal{F}$ per dato un problema di classificazione sia così grande rispetto al training set che è spesso possibile trovare due funzioni $f, f' \in \mathcal{F}$ che classificano bene gli esempi del training set (cioè hanno entrambe training error vicino a zero) ma tali che f predice gli elementi del test set molto meglio di f'. In questo caso, un algoritmo di apprendimento non riuscirà a scegliere il classificatore buono f scartando quello cattivo f' in quanto i due classificatori sono pressoché indistinguibili sul training set. Per evitare che ciò avvenga, il training set dovrebbe essere abbastanza grande per riuscire a discriminare le f buone da quelle cattive.